IN THE CLAIMS:

1-45. (Cancel)

46. (New) A system for synchronizing a cached file with a database:

a computer processor;

a network connection device operable to establish a connection with a database;

a computer readable memory containing a local cache; and

a software program, executable to run in user space on a client computer, stored on a computer medium and executable by the computer processor to:

send a request to the database for a file;

receive the file at the client computer directly from a database;

store the file as a cached file in the local cache;

notify the operating system to open the cached file using a locally running application associated with the file type for the cached file;

determine if the cached file at the client computer has been modified by a user using the locally running application based on a notification from a file management system of an operating system; and

if the cached file has been modified, save the cached file from the cache directly to the database.

47. (New) The system of Claim 46, further comprising an operating system operable to open the cached file the application associated with a file type for the cached file.

48. (New) The system of Claim 46, wherein the software program is further executable to associate the cached file with a connection, wherein the connection is associated with the database.

49. (New) The system of Claim 48, wherein the software program is further operable to establish the connection.

50. (New) The system of Claim 49, wherein the software program is further executable to:

determine if the connection has been disconnected; and

if the connection has been disconnected, to reestablish the connection.

51. (New) The system of Claim 49, wherein the software program is further executable to:

save a user login; and

reestablish the connection using the user login.

52. (New) The system of Claim 46, wherein the software program is further executable to associate the cached file with a location in the cache.

53. (New) The system of Claim 52, wherein an application accessing the cached file saves the cached file at the location in the cache associated with the cached file.

54. (New) The system of Claim 47, wherein the software program is further executable to receive a database notification from a database management program that an additional user has modified the database asset.

55. (New) The system of Claim 54, wherein the software program is further executable to provide a notice to a first user that the additional user has modified the database asset.

56. (New) The system of Claim 55, wherein the software program is further executable to provide the first user an option of overriding a modification to the database asset made by the additional user.

57. (New) The system of Claim 46, wherein the software program is further executable to:

receive the request from a first user for the database asset.

58. (New) The system of Claim 46, wherein the software program is executable to:

receive a notification that said database asset has been deleted from said database; and

purge said cached file from said cache.

59. (New) A system for synchronizing a file in a cache comprising:

a database server further comprising:

a server processor;

a server memory; a database stored on said server memory containing a plurality of files; and

a database management program stored on the server memory executable by the server processor to:

receive a client request for a file; and

retrieve the file; and

a client computer communication with the database server further comprising:

a client processor;

a client memory; and

a cache manager program, stored on the client memory executable by the client processor to run in user space and to:

establish a connection to the database server;

forward the client request for the file to the database server;

receive the file from the database server;

store the file as a cached file in the client memory;

notify an operating system to open the cached file with a locally running application associated with a file type for the cached file;

determine if the cached file has been modified based on a notification from a file management system of an operating system; and

if the cached file has been modified, communicate the cached file to the database.

60. (New) The system of Claim 59, wherein the cache manager program is further executable to prompt the operating system to access the cached file using the application.

61. (New) The system of Claim 60, wherein the operating system further comprises a file management system.

62. (New) The system of Claim 59, wherein the cache management program is further executable to associate the cached file with a connection, wherein the connection is associated with the database.

63. (New) The system of Claim 62, wherein the cache management program is further executable to:

determine if the connection has been disconnected; and

if the connection has been disconnected, to re-establish the connection.

64. (New) The system of Claim 63, wherein the cache management program is further executable to:

save a user login; and

reestablish the connection using the user login.

65. (New) The system of Claim 59, wherein the cache management program is further executable to associate the cached file with a location in the cache.

66. (New) The system of Claim 65, wherein an application accessing the cached file saves the cached file at the location in the cache associated with the cached file.

67. (New) The system of Claim 59, wherein the database management program is executable to notify the client computer if an additional client modifies the file, and wherein the cache manager program is executable to receive the notification from the database management program.

68. (New) The system of Claim 67, wherein the cache management program is further executable to provide a warning to a first user that the additional user has modified the file.

69. (New) The system of Claim 59, wherein the database management program is executable to notify the client computer that the file has been deleted from the database, and wherein the cache manager is operable to purged the cached file from the cache.

70. (New) A method for synchronizing a file in a cache comprising:

receiving a database asset directly from a database;

storing the database asset in a cache as a cached file;

opening the cached file with a local application associated with a file type for the cached file;

determining if the cached file has been modified based on a notification from a file management system of an operating system; and

if the cached file has been modified, communicating the cached file directly to the database.

71. (New) The method of claim 70, further comprising associating the cached file with a connection.

72. (New) The method of Claim 71, further comprising establishing the connection with the database.

73. (New) The method of Claim 72, further comprising:

determining if the connection with the database has become disconnected; and

if the connection with the database has become disconnected, reestablishing the connection to the database.

74. (New) The method of Claim 73, further comprising saving a user login and using the saved user login to reestablish the connection.

75. (New) The method of Claim 70, further comprising associating the cached file with a location in a memory.

76. (New) The method of Claim 70, further comprising notifying a first user that an additional user has accessed the database asset.

77. (New) The method of Claim 70 further comprising opening the cached file with an application associated with a file type associated with the cached file.

78. (New) The method of claim 70 further comprising purging the cached file from the cache if the database asset is deleted from the database.

79. (New) A method of managing a cache comprising:

establishing a connection with a database;

retrieving a database asset from the database;

storing the database asset in a cache as a cached file;

associating the cached file with the connection;

opening the cached file in local application associated with the file type for the cached file;

determining if the cached file has been modified based on a notification from a file management system of an operating system;

if the cached file has been modified:

determining if the connection with the database is still established;

if the connection with the database is not still established, reestablishing the connection to the database;

communicating the cached file directly to the database; and

saving the cached file in the database as the database asset.

80. (New) A system for synchronizing a cached file with a database:

a computer processor;

a network connection device operable to establish a connection with a database;

a computer readable memory containing a cache; and

a software program, executable to run in user space, stored on the computer readable memory and executable by the computer processor to:

send a request to the database for a database asset;

receive a database asset directly from a database;

notify an operating system to open the cached file with a local application associated with the file type for the cached file;

store the database asset as a cached file in the cache;

determine if the cached file has been modified by polling the cached asset according to a predefined schedule; and

if the cached file has been modified, save the cached file directly to the database.

81. (New) A system for synchronizing a cached file with a database:

a computer processor;

a network connection device operable to establish a connection with a database;

a computer readable memory containing a local cache; and

a software program, executable to run in user space, stored on a computer medium and executable by the computer processor to:

send a request to the database for a first database asset;

receive the first database asset directly from a database;

store the first database asset as a first cached file in the local cache;

notify the operating system to open the first cached file using a first locally running application associated with the file type for the cached file;

determine if the first cached file has been modified by the first locally running application based on a notification from a file management system of an operating system; and

if the first cached file has been modified, save the cached from the cache the database;

send a request to the database for a second database asset;

receive the second database asset directly from the database;

store the second database asset as a second cached file in the local cache;

notify the operating system to open the second cached file using a second locally running application associated with the file type for the cached file;

determine if the second cached file has been modified by the second locally running application based on a notification from a file management system of an operating system; and

if the second cached file has been modified, save the cached file directly to the database, wherein the first database asset and the second database asset have different file types and the first locally running application and the second locally running application are different application types.